<https://help.sap.com/docs/SUPPORT_CONTENT/bwdabc/3361384613.html>

2LIS\_03\_BF: BWVORG / BWBREL / BWMNG / BWGEO / BWGVO / BWGVP

Regarding the meaning of different BWVORG, please refer Note [**492828**- Determining the transaction key for 2LIS\_03\_BF + 2LIS\_03\_UM](https://launchpad.support.sap.com/#/notes/492828).

The detailed logic can be seen in: FORM ANREICHERN\_MSEG\_COR / inc LMCRSF03 / in prg LMCRSF03

  XMCMSEG-**BWVORG** = CON\_BWVORG\_NONE.  -- defaul: 499

*\* Kennzeichen setzen, ob Transfersatz bestandsrelevant ist*

*\* Bei Wertartikelbuchungen ist nur der gedoppelte Satz relevant!!*

\* Set whether the transfer rate is relevant for inventory

\* For value-added item postings, only the duplicate rate is relevant!

  IF XMCMSEG-MATNR EQ XMCMSEG-MATBF OR *"-> bestandsrelevant* "= stock relevant

     XMCMSEG-MATBF IS INITIAL.         *" -> Sicherheitshalber*

    XMCMSEG-**BWBREL**= CON\_STOCK\_RELEVANT.     *"****=1***

  ELSE.                                *"keine Relevanz für Bestand    "=*no relevance for stock

    XMCMSEG-**BWBREL**= CON\_NOT\_STOCK\_RELEVANT. *"****=2***

  ENDIF.

*\* Basismengeneinheiten ungleich Erfassungsmengeneinheit*

*\* Doppelung von Sätzen tritt nur im Retail auf und wird bereits im*

*\* BCO durchgeführt -> Doppelungsflag wurde gesetzt*.

\* Base units of measure do not equal the recording unit of measure

\* Duplicate records only occur in retail and are already carried out in

\* BCO -> Duplicate flag has been set.

  IF XMCMSEG-DPFLG EQ AKTIV.           *"Warengrp. Wertartikel "Goods group: Valuable items*

    IF XMCMSEG-MEINS EQ XMCMSEG-ERFME.

      HLP\_MENGE = XMCMSEG-ERFMG.

    ELSE.

      PERFORM [UNIT\_CONVERSION](#_UNIT_CONVERSION)USING XMCMSEG-MATNR

              XMCMSEG-ERFME XMCMSEG-MEINS XMCMSEG-ERFMG

              CHANGING  HLP\_MENGE.

    ENDIF.

  ENDIF.

*\* Allgemeine Verarbeitung der MCMSEG*

*\* Mengen-Verarbeitung auf Basis der Hilfsmenge*

\* General processing of the MCMSEG

\* Quantity processing based on the auxiliary quantity

  IF HLP\_MENGE IS INITIAL.             "*Umrechnungsmenge*?? Conversion amount??

    HLP\_MENGE = XMCMSEG-MENGE.         "*Menge uebernehmen Accept quantity*

  ENDIF.

...

*\* Wareneingänge direkt in den Verbrauch sind nicht bestandsrelevant*

*\* Bestandstyp = 'V' und Bestandsausprägung = 'V'.*

\* Goods receipts directly into consumption are not inventory-relevant.

\* Stock type = 'V' and stock characteristic = 'V'.

  IF XMCMSEG-BSTTYP  = CON\_BSTTYP\_VERBRAUCH AND

     XMCMSEG-BSTAUS = CON\_BSTAUS\_VERBR.

    XMCMSEG-BWBREL = CON\_NOT\_STOCK\_RELEVANT.

  ENDIF.

*\* Transfermengen/-Werte immer positive \* Transfer quantities/values ​​always positive*

  XMCMSEG-**BWMNG**  = HLP\_MENGE.

  XMCMSEG-**BWGEO**  = XMCMSEG-DMBTR.

  XMCMSEG-**BWGVO**= XMCMSEG-VKWRA.

  XMCMSEG-**BWGVP**= XMCMSEG-VKWRT.

*\* Zu-/Abgänge sind über die Variale ZU\_ABGANG erkennbar: Incomings/outgoings can be identified via the variable IN\_ABGANG:*

*\* ZU\_ABGANG < 0 -> ABGANG (-1)*

*\* ZU\_ABGANG > 0 -> ZUGANG (+1)*

  PERFORM FUELLEN\_ZUABGANG\_MSEG CHANGING XMCMSEG

                                         ZU\_ABGANG.

  IF ZU\_ABGANG GT 0.                   *"Zugang*

    MOVE '**000**' TO XMCMSEG-**BWVORG**.

  ELSE.

    MOVE '**100**' TO XMCMSEG-**BWVORG**.      *"Abgang*

  ENDIF.

*...*

*\*.....Pruefen ob Bestandskorrekturen Inventur vorliegen Check whether inventory corrections are available*

  IF XMCMSEG-INVKZ EQ AKTIV.           *"Inventur maschinell*“= corresponds to a physical inventory, from TMCA-INVKZ

    IF ZU\_ABGANG GT 0.

*\*.....Bestandskorrektur Inventur PLUS Inventory correction inventory*

      MOVE '**005**' TO XMCMSEG-**BWVORG**.  "= Stock adjustment inventory + Receipt

    ELSE.

*\*.....Bestandskorrektur Inventur Minus Inventory correction inventory*

      MOVE '**105**' TO XMCMSEG-**BWVORG**. ”= Stock adjustment inventory - Issue

    ENDIF.

    EXIT.

  ENDIF.

*\*.....Pruefen ob Bestandskorrekturen Sonstige vorliegen Check whether other inventory adjustments are available*

  IF XMCMSEG-KORR  EQ AKTIV.           *"Bestandskorrekturen*

    IF ZU\_ABGANG GT 0.

*\*.....Bestandskorrektur Sontige PLUS Inventory adjustment Other*

      MOVE '**006**' TO XMCMSEG-**BWVORG**.  "= Stock adjustment other + Receipt

    ELSE.

*\*.....Bestandskorrektur Sonstige Minus*"= stock correction

      MOVE '**106**' TO XMCMSEG-**BWVORG**.  “= Stock adjustment inventory - Issue

    ENDIF.

    EXIT.

  ENDIF.

*\*.....Pruefen Wareneingang/Retoure  "*= Check goods receipt/return

  IF FLG\_LIEF\_GEFUELLT EQ TRUE.

    IF ZU\_ABGANG > 0.

*\*.....Wareneingang /Lieferant Goods receipt / supplier*

      MOVE '**001**' TO XMCMSEG-**BWVORG**.  "= Goods receipt

    ELSE.

*\*.....Retoure Lieferant Return supplier*

      MOVE '**101**' TO XMCMSEG-**BWVORG**.  "= Returns

    ENDIF.

    EXIT.

  ENDIF.

# BWBREL:

<https://help.sap.com/docs/SUPPORT_CONTENT/bwdabc/3361384477.html>

It has 2 values: 1 and 2.

1 - stock relevant

2 - stock not relevant

FORM ANREICHERN\_MSEG\_CORE  
in prg LMCRSF03, inc LMCRSF03

IF XMCMSEG-MATNR EQ XMCMSEG-MATBF OR "-> bestandsrelevant  
     XMCMSEG-MATBF IS INITIAL. " -> Sicherheitshalber  
     XMCMSEG-**BWBREL** = CON\_STOCK\_RELEVANT. "**=1**  
ELSE. "keine Relevanz für Bestand  
     XMCMSEG-**BWBREL** = CON\_NOT\_STOCK\_RELEVANT. "**=2**  
ENDIF.

*\* Wareneingänge direkt in den Verbrauch sind nicht bestandsrelevant  
\* Bestandstyp = 'V' und Bestandsausprägung = 'V'. Goods receipts directly into consumption are not inventory-relevant.\* Inventory type = 'V' and inventory characteristic = 'V'.*

  IF XMCMSEG-BSTTYP  = CON\_BSTTYP\_VERBRAUCH AND  
     XMCMSEG-BSTAUS = CON\_BSTAUS\_VERBR.  
    XMCMSEG-**BWBREL** = CON\_NOT\_STOCK\_RELEVANT.  "**=2**  
  ENDIF.

## UNIT\_CONVERSION

*\*\*&---------------------------------------------------------------------*  
*\**  
*\*&      Form  UNIT\_CONVERSION*  
*\*&---------------------------------------------------------------------\**  
*\*       Umrechnen Erfassungsmenge in Basismenge                        \**  
*\*----------------------------------------------------------------------\**  
*\*  -->  x\_matnr   Materialnummer*  
*\*  -->  x\_erfme   Erfassungsengeneinheit*  
*\*  -->  x\_basme   Basismengeneinheit*  
*\*  -->  x\_erfmg   Menge In Erfassungsengeneinheit*  
*\*  <--  x\_basmg   Menge In Basismengeneinheit*  
*\*----------------------------------------------------------------------\**  
FORM UNIT\_CONVERSION USING X\_MATNR X\_ERFME X\_BASME X\_ERFMG  
                  CHANGING X\_BASMG .  
  
  CALL FUNCTION 'MB\_UNIT\_CONVERSION'                          *"#EC FB\_RC*  
       EXPORTING  
            ERFME                = X\_ERFME  
            ERFMG                = X\_ERFMG  
            MATNR                = X\_MATNR  
            MEINS                = X\_BASME  
       IMPORTING  
            O\_MENGE              = X\_BASMG  
       EXCEPTIONS  
            CONVERSION\_NOT\_FOUND = 1  
            DIFFERENT\_DIMENSION  = 2  
            DIVISION\_BY\_ZERO     = 3  
            OVERFLOW             = 4  
            T006D\_ENTRY\_MISSING  = 5  
            T006\_ENTRY\_MISSING   = 6  
            OTHERS               = 7.  
ENDFORM.                               *" UNIT\_CONVERSION*

## MB\_UNIT\_CONVERSION (function module in SE37)

FUNCTION MB\_UNIT\_CONVERSION.  
*\*"----------------------------------------------------------------------*  
*\*"\*"Lokale Schnittstelle:*  
*\*"  IMPORTING*  
*\*"     VALUE(ERFME) LIKE  MSEG-ERFME DEFAULT SPACE*  
*\*"     VALUE(ERFMG) LIKE  MSEG-ERFMG DEFAULT 0*  
*\*"     VALUE(MATNR) LIKE  MARA-MATNR DEFAULT SPACE*  
*\*"     VALUE(MEINS) LIKE  MARA-MEINS DEFAULT SPACE*  
*\*"     VALUE(MENGE) LIKE  MSEG-MENGE DEFAULT 0*  
*\*"     VALUE(UMREN) DEFAULT 1*  
*\*"     VALUE(UMREZ) DEFAULT 0*  
*\*"     VALUE(CHARG) LIKE  MSEG-CHARG OPTIONAL*  
*\*"     VALUE(WERKS) LIKE  MSEG-WERKS OPTIONAL*  
*\*"     VALUE(CUOBJ) LIKE  INOB-CUOBJ OPTIONAL*  
*\*"  EXPORTING*  
*\*"     VALUE(O\_ERFME) LIKE  MSEG-ERFME*  
*\*"     VALUE(O\_ERFMG) LIKE  MSEG-ERFMG*  
*\*"     VALUE(O\_MEINS) LIKE  MARA-MEINS*  
*\*"     VALUE(O\_MENGE) LIKE  MSEG-MENGE*  
*\*"     VALUE(O\_UMREN)*  
*\*"     VALUE(O\_UMREZ)*  
*\*"  EXCEPTIONS*  
*\*"      CONVERSION\_NOT\_FOUND*  
*\*"      DIFFERENT\_DIMENSION*  
*\*"      DIVISION\_BY\_ZERO*  
*\*"      OVERFLOW*  
*\*"      T006D\_ENTRY\_MISSING*  
*\*"      T006\_ENTRY\_MISSING*  
*\*"----------------------------------------------------------------------*  
*\*- Default Werte setzten ----------------------------------------------\**  
  O\_ERFMG = ERFMG.                     *"Keine Umrechnung erforderlich*  
  O\_MENGE = MENGE.  
  O\_MEINS = MEINS.  
  O\_ERFME = ERFME.  
  O\_UMREZ = UMREZ.  
  O\_UMREN = UMREN.  
*\* See note in note 623071*  
*\* YMENGE is the given quantity (this could be either ERFMG or MENGE*  
*\* depending on XKZMEINH)*  
*\* XMENGE is the calculated result*  
*\* XKZMEINH = SPACE   MENGE -> ERFMG*  
*\* XKZMEINH = X       ERFMG -> MENGE*  
  IF MENGE = 0 AND NOT ERFMG = 0.  
    XMENGE = ERFMG.  
    YMENGE = ERFMG.  
    XKZMEINH = 'X'.                    *"Umzurech. Menge auf AME*  
  ELSEIF NOT MENGE = 0 AND ERFMG = 0.  
    XMENGE = MENGE.  
    YMENGE = MENGE.  
    XKZMEINH = SPACE.                  *"Umzurech. Menge auf die LME*  
  ELSE.  
    EXIT.  
  ENDIF.  
  IF ERFME = MEINS.  
    UMREN = UMREZ = 1.  
    O\_ERFMG = YMENGE.                  *"Umgerechnete Menge*  
    O\_MENGE = XMENGE.                  *"Eingegebene  Menge*  
    O\_UMREZ = UMREZ.  
    O\_UMREN = UMREN.  
    EXIT.  
  ENDIF.  
  
  DATA: L\_BATCH\_CONV\_ERROR TYPE C,  
        LF\_FRACT5\_CONVERSION TYPE C VALUE 'X',  
        LC\_NO\_FACTOR       TYPE C VALUE '1',  
        L\_UMREN TYPE I,  
        L\_UMREZ TYPE I,  
        L\_KZWSO LIKE MARM-KZWSO,  
        T       TYPE C,  
        L\_UMREZ\_F TYPE F,  
        L\_UMREN\_F TYPE F,  
        LT\_DUMMY\_MARM TYPE STANDARD TABLE OF MARM.  
ENHANCEMENT-POINT MB\_UNIT\_CONVERSION\_01 SPOTS ES\_SAPLMAME STATIC.  
*\*$\*$-Start: MB\_UNIT\_CONVERSION\_01---------------------------------------------------------------$\*$\**  
ENHANCEMENT 1  /SAPMP/PIECEBATCH\_MB\_UNIT\_C000.    *"active version*  
*\*XFO 13.08.03 H650868 wrong proposal of the amount*  
data:    lc\_VALUE\_0         type c value '2'.  
ENDENHANCEMENT.  
*\*$\*$-End:   MB\_UNIT\_CONVERSION\_01---------------------------------------------------------------$\*$\**  
  
*\* Falls Charge nicht initial*  
  IF NOT CHARG IS INITIAL.  
*\* Prüfe, ob variable/Anteilsabwicklung aktiv für Mat und Meinh*  
  CALL FUNCTION 'VBWS\_UNIT\_CHECK\_FOR\_MATNR'  
     EXPORTING  
          I\_MATNR             = MATNR  
          I\_MEINH             = ERFME  
          WITH\_UNITS\_OF\_SAME\_DIMENSION = 'X'              *"note 943348*  
     EXCEPTIONS  
          OTHERS              = 4.  
*\* Falls variable/Anteilsabwicklung aktiv für Mat und Meinh*  
    IF SY-SUBRC IS INITIAL.  
      CALL FUNCTION 'MATERIAL\_UNIT\_CONVERSION'  
           EXPORTING  
                KZMEINH                = XKZMEINH  
                MATNR                  = MATNR  
                MEINH                  = ERFME  
                MEINS                  = MEINS  
                NO\_OUTPUT              = 'X'  
                CHARGE                 = CHARG  
                WERKS                  = WERKS  
           IMPORTING  
                UMREN                  = L\_UMREN  
                UMREZ                  = L\_UMREZ  
                KZWSO                  = L\_KZWSO  
                BATCH\_CONVERSION\_ERROR = L\_BATCH\_CONV\_ERROR  
           EXCEPTIONS  
                CONVERSION\_NOT\_FOUND = 1  *"Umrechnungsf. nicht bestimmt*  
                OVERFLOW             = 2  *"Feldüberlauf*  
                MATERIAL\_NOT\_FOUND   = 3  *"Material existiert nicht*  
                MEINS\_MISSING        = 4  *"Keine BasismE vorhaNDEN*  
                NO\_MEINH             = 5  *"Keine AlME angegeben*  
                OUTPUT\_INVALID       = 6  *"Menge ist kein Zahl*  
                MEINH\_NOT\_FOUND      = 7. *"ALME nicht zulässig*  
      CASE SY-SUBRC.  
        WHEN 1.  
         MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen nicht mgl.*  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW.  *"                 Feldüberlauf*  
        WHEN 3.  
         MESSAGE E103 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 4.  
          MESSAGE E104 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 5.  
          MESSAGE E108 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 6.  
         MESSAGE E102 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 7.  
          MESSAGE E107 RAISING CONVERSION\_NOT\_FOUND.  
      ENDCASE.  
*\* Umrechnungsfaktoren aus Charge liegen vor*  
      IF SY-SUBRC IS INITIAL.  
*\* Prüfe, ob Flag XFHDW für irgendeine ME sitzt*  
        CALL FUNCTION 'VBWS\_UNITS\_OF\_MATERIAL'  
          EXPORTING  
            i\_matnr                            = matnr  
            LEADING\_UNIT                       = 'X'  
          tables  
            ws\_marm\_tab                        = lt\_dummy\_marm  
          EXCEPTIONS  
            NO\_UNITS                           = 2  
            UNITS\_NOT\_ACTIVE                   = 3  
            OTHERS                             = 4.  
*\* Umrechnungsfaktoren aus Charge verwenden, wenn XFHDW nicht gesetzt*  
*\* oder Material mit Anteil/Wirkstoffmengen*  
ENHANCEMENT-SECTION     MB\_UNIT\_CONVERSION\_02 SPOTS ES\_SAPLMAME.  
        IF L\_BATCH\_CONV\_ERROR = LC\_NO\_FACTOR AND  
           L\_KZWSO            = 'B'          AND  
           SY-SUBRC           = 0            AND  
           NOT  UMREZ IS INITIAL             AND  
           NOT  UMREN IS INITIAL.  
*\* Use factors from caller = do nothing*  
        ELSE.  
*\* Use factors from batch = move L\_UMREZ to UMREN and L\_UMREN to UMREN*  
        IF  L\_KZWSO CA 'AB' AND  
            L\_BATCH\_CONV\_ERROR IS INITIAL.  
*\*         Determine whether conversion needs to be better than fract5*  
          CALL FUNCTION 'MURC\_GET\_CONVERSION\_TYPE'  
            IMPORTING  
              EF\_FRACT5\_CONVERSION  = LF\_FRACT5\_CONVERSION.  
        endif.  
        DESCRIBE FIELD UMREZ TYPE T.  
        IF T EQ 'P'.  
          L\_UMREZ\_F = L\_UMREZ.  
          L\_UMREN\_F = L\_UMREN.  
          CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
            EXPORTING  
                NOMIN               = L\_UMREZ\_F  
                DENOMIN             = L\_UMREN\_F  
            IMPORTING  
                NOMOUT              = \*MARM-UMREZ  
                DENOMOUT            = \*MARM-UMREN  
            EXCEPTIONS  
                CONVERSION\_OVERFLOW = 1.  
          IF SY-SUBRC = 0.  
            UMREZ = \*MARM-UMREZ.  
            UMREN = \*MARM-UMREN.  
          ELSE.  
            MESSAGE E037(BM) RAISING OVERFLOW.  
          ENDIF.  
        ELSE.  
          UMREN = L\_UMREN.  
          UMREZ = L\_UMREZ.  
        ENDIF.  
        ENDIF. *"Umrechnungsfaktoren aus Charge verwenden ?*  
END-ENHANCEMENT-SECTION.  
*\*$\*$-Start: MB\_UNIT\_CONVERSION\_02---------------------------------------------------------------$\*$\**  
ENHANCEMENT 2  /SAPMP/PIECEBATCH\_MB\_UNIT\_C000.    *"active version*  
*\*XFO 13.08.03 H650868 wrong proposal of the amount*  
        IF ( L\_BATCH\_CONV\_ERROR = LC\_NO\_FACTOR or  
*\* characteristic with value 0 is handled, as not set (no factor)*  
             L\_BATCH\_CONV\_ERROR = LC\_value\_0 ) and  
           L\_KZWSO            = 'B'          AND  
           SY-SUBRC           = 0            AND  
           NOT  UMREZ IS INITIAL             AND  
           NOT  UMREN IS INITIAL.  
*\* Use factors from caller = do nothing*  
        ELSE.  
*\* Use factors from batch = move L\_UMREZ to UMREN and L\_UMREN to UMREN*  
        IF  L\_KZWSO CA 'AB' AND  
            L\_BATCH\_CONV\_ERROR IS INITIAL.  
*\*         Determine whether conversion needs to be better than fract5*  
          CALL FUNCTION 'MURC\_GET\_CONVERSION\_TYPE'  
            IMPORTING  
              EF\_FRACT5\_CONVERSION  = LF\_FRACT5\_CONVERSION.  
        endif.  
        DESCRIBE FIELD UMREZ TYPE T.  
        IF T EQ 'P'.  
          L\_UMREZ\_F = L\_UMREZ.  
          L\_UMREN\_F = L\_UMREN.  
          CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
            EXPORTING  
                NOMIN               = L\_UMREZ\_F  
                DENOMIN             = L\_UMREN\_F  
            IMPORTING  
                NOMOUT              = \*MARM-UMREZ  
                DENOMOUT            = \*MARM-UMREN  
            EXCEPTIONS  
                CONVERSION\_OVERFLOW = 1.  
          IF SY-SUBRC = 0.  
*\* Mill 0024 Single Unit Batch SW*  
        Data: lf\_use\_batch\_factors type xfeld.  
        lf\_use\_batch\_factors = 'X'.  
          ELSE.  
            MESSAGE E037(BM) RAISING OVERFLOW.  
          ENDIF.  
        ELSE.  
*\* Mill 0024 Single Unit Batch SW*  
        lf\_use\_batch\_factors = 'X'.  
        ENDIF.  
        ENDIF. *"Umrechnungsfaktoren aus Charge verwenden ?*  
  
ENDENHANCEMENT.  
*\*$\*$-End:   MB\_UNIT\_CONVERSION\_02---------------------------------------------------------------$\*$\**  
      ENDIF. *"Umrechnungsfaktoren aus Charge liegen vor ?*  
    ENDIF. *"variable/Anteilsabwicklung aktiv ?*  
  ENDIF. *"Charge ?*  
  
*\*------------- Materialnummer nicht vorhanden ------------------------\**  
*\*  if ( ( umren is initial or umrez is initial ) or*  
*\*    ( not charg is initial and sy-subrc is initial ) ).  "SW ALRK WS-St*  
  
ENHANCEMENT-SECTION     MB\_UNIT\_CONVERSION\_03 SPOTS ES\_SAPLMAME.  
  IF ( UMREN IS INITIAL OR UMREZ IS INITIAL ) OR  
       LF\_FRACT5\_CONVERSION IS INITIAL.  
    IF MATNR IS INITIAL.               *"Keine Materialnummer*  
      IF XKZMEINH IS INITIAL.          *"Umzurechnende Menge in LME*  
*\*- Umrechnungsfaktor(en) sind nicht gesetzt Materialnummer nicht vorh.-\**  
        CALL FUNCTION 'UNIT\_CONVERSION\_SIMPLE'  
             EXPORTING  
                  INPUT                = XMENGE  
                  NO\_TYPE\_CHECK        = SPACE  *"Typenprüfung erfolgt*  
                  UNIT\_IN              = MEINS  
                  UNIT\_OUT             = ERFME  
             IMPORTING  
                  ADD\_CONST            = ADDK  
                  DENOMINATOR          = UMREZ\_GES  
                  NUMERATOR            = UMREN\_GES  
                  OUTPUT               = XMENGE  
             EXCEPTIONS  
                  CONVERSION\_NOT\_FOUND = 1  
                  OVERFLOW             = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen unmögl.*  
          WHEN 2.  
            MESSAGE E106 RAISING OVERFLOW.   *"Feldüberlauf*  
        ENDCASE.  
  
    CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
         EXPORTING  
              NOMIN               =  UMREN\_GES  
              DENOMIN             =  UMREZ\_GES  
         IMPORTING  
              NOMOUT              =  UMREN\_DEC  
              DENOMOUT            =  UMREZ\_DEC  
         EXCEPTIONS  
              CONVERSION\_OVERFLOW = 1  
              OTHERS              = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E106 RAISING OVERFLOW.   *"Feldüberlauf*  
          WHEN 2.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen unmögl.*  
        ENDCASE.  
  
        O\_ERFMG = XMENGE.              *"623071*  
        O\_MENGE = YMENGE.              *"623071*  
        O\_ERFME = MEINS.  
        O\_UMREZ = UMREZ\_DEC.  
        O\_UMREN = UMREN\_DEC.  
      ELSE.                            *"Umzurechnende Menge in AME*  
        CALL FUNCTION 'UNIT\_CONVERSION\_SIMPLE'  
             EXPORTING  
                  INPUT                = XMENGE  
                  NO\_TYPE\_CHECK        = SPACE  *"Typenprüfung erfolgt*  
                  UNIT\_IN              = ERFME  
                  UNIT\_OUT             = MEINS  
             IMPORTING  
                  ADD\_CONST            = ADDK  
                  DENOMINATOR          = UMREZ\_GES  
                  NUMERATOR            = UMREN\_GES  
                  OUTPUT               = XMENGE  
             EXCEPTIONS  
                  CONVERSION\_NOT\_FOUND = 1  
                  OVERFLOW             = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrech. unmögl.*  
          WHEN 2.  
            MESSAGE E106 RAISING OVERFLOW.          *"Feldüberlauf*  
        ENDCASE.  
    CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
         EXPORTING  
              NOMIN               =  UMREN\_GES  
              DENOMIN             =  UMREZ\_GES  
         IMPORTING  
              NOMOUT              =  UMREN\_DEC  
              DENOMOUT            =  UMREZ\_DEC  
         EXCEPTIONS  
              CONVERSION\_OVERFLOW = 1  
              OTHERS              = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E106 RAISING OVERFLOW.   *"Feldüberlauf*  
          WHEN 2.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen unmögl.*  
        ENDCASE.  
        O\_ERFMG = YMENGE.              *"Umgerechnete Menge*  
        O\_MENGE = XMENGE.              *"Eingegebene  Menge*  
        O\_ERFME = MEINS.  
        O\_UMREZ = UMREZ\_DEC.  
        O\_UMREN = UMREN\_DEC.  
      ENDIF.  
    ELSE.  
*\*-- Materialnummer vorhanden Umrechnungsfaktor(en) sind nicht gesetzt--\**  
*\*-- KZMEINH Umrechnung in LME oder AME --------------------------------\**  
      CALL FUNCTION 'MATERIAL\_UNIT\_CONVERSION'  
           EXPORTING  
                INPUT                = XMENGE  
                KZMEINH              = XKZMEINH  
                MATNR                = MATNR  
                CHARGE               = CHARG  
                WERKS                = WERKS  
                MEINH                = ERFME  
                MEINS                = MEINS  
           IMPORTING  
                OUTPUT               = XMENGE  
           EXCEPTIONS  
                CONVERSION\_NOT\_FOUND = 1  *"Umrechnungsf. nicht bestimmt*  
                OVERFLOW             = 2  *"Feldüberlauf*  
                MATERIAL\_NOT\_FOUND   = 3  *"Material existiert nicht*  
                MEINS\_MISSING        = 4  *"Keine BasismE vorhaNDEN*  
                NO\_MEINH             = 5  *"Keine AlME angegeben*  
                OUTPUT\_INVALID       = 6  *"Menge ist kein Zahl*  
                MEINH\_NOT\_FOUND      = 7. *"ALME nicht zulässig*  
      CASE SY-SUBRC.  
        WHEN 1.  
         MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen nicht mgl.*  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW.  *"                 Feldüberlauf*  
        WHEN 3.  
         MESSAGE E103 RAISING  CONVERSION\_NOT\_FOUND.  
        WHEN 4.  
          MESSAGE E104 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 5.  
          MESSAGE E108 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 6.  
         MESSAGE E102 RAISING  CONVERSION\_NOT\_FOUND.  
        WHEN 7.  
          MESSAGE E107 RAISING CONVERSION\_NOT\_FOUND.  
      ENDCASE.  
      IF XKZMEINH IS INITIAL.  
       O\_ERFMG = XMENGE.                *"Eingegebene  Menge*  
       O\_MENGE = YMENGE.                *"Umgerechnete Menge*  
      ELSE.  
       O\_ERFMG = YMENGE.                *"Umgerechnete Menge*  
       O\_MENGE = XMENGE.                *"Eingegebene  Menge*  
      ENDIF.  
  
       O\_ERFME = ERFME.  
       describe field o\_umrez type t.  
       IF T EQ 'P'.  
         L\_UMREZ\_F = UMREZ.  
         L\_UMREN\_F = UMREN.  
         CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
           EXPORTING  
               NOMIN               = L\_UMREZ\_F  
               DENOMIN             = L\_UMREN\_F  
           IMPORTING  
               NOMOUT              = \*MARM-UMREZ  
               DENOMOUT            = \*MARM-UMREN  
           EXCEPTIONS  
               CONVERSION\_OVERFLOW = 1.  
         IF SY-SUBRC = 0.  
           O\_UMREZ = \*MARM-UMREZ.  
           O\_UMREN = \*MARM-UMREN.  
         ELSE.  
           MESSAGE E037(BM) RAISING OVERFLOW.  
         ENDIF.  
       ELSE.  
         O\_UMREZ = UMREZ.  
         O\_UMREN = UMREN.  
       ENDIF.  
  
      IF UMREN  = UMREZ.               *"Umrechnungsfaktoren gleich ?*  
        EXIT.                          *"Raus*  
      ENDIF.  
    ENDIF.  
  ELSE.  
*\*----- Umrechnungsfaktor(en) ist/sind gesetzt -------------------------\**  
    IF XKZMEINH IS INITIAL.            *"Umzurech. Menge in LME*  
*\*-------- Menge in Lagermengeneinheit umrechnen ----------------------\**  
      CALL FUNCTION 'UNIT\_CONVERSION\_WITH\_FACTOR'  
           EXPORTING  
                ADD\_CONST   = ADDK  
                DENOMINATOR = UMREZ  
                INPUT       = XMENGE  
                NUMERATOR   = UMREN  
           IMPORTING  
                OUTPUT      = XMENGE  
           EXCEPTIONS  
                OVERFLOW    = 2.  
      CASE SY-SUBRC.  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW. *"Feldüberlauf*  
      ENDCASE.  
      O\_ERFMG = XMENGE.                *"Umgerechnete Menge*  
      O\_MENGE = YMENGE.                *"Eingegebene  Menge*  
      O\_ERFME = ERFME.  
*\*     note 1158817: avoid dump BCD\_FIELD\_OVERFLOW*  
*\*      O\_UMREZ = UMREZ.*  
*\*      O\_UMREN = UMREN.*  
      describe field o\_umrez type t.  
      IF T EQ 'P'.  
        L\_UMREZ\_F = UMREZ.  
        L\_UMREN\_F = UMREN.  
        CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
          EXPORTING  
              NOMIN               = L\_UMREZ\_F  
              DENOMIN             = L\_UMREN\_F  
          IMPORTING  
              NOMOUT              = \*MARM-UMREZ  
              DENOMOUT            = \*MARM-UMREN  
          EXCEPTIONS  
              CONVERSION\_OVERFLOW = 1.  
        IF SY-SUBRC = 0.  
          O\_UMREZ = \*MARM-UMREZ.  
          O\_UMREN = \*MARM-UMREN.  
        ELSE.  
          MESSAGE E037(BM) RAISING OVERFLOW.  
        ENDIF.  
      ELSE.  
        O\_UMREZ = UMREZ.  
        O\_UMREN = UMREN.  
      ENDIF.  
    ELSE.                              *"Umzurechnende Menge in AME*  
*\*-------- Menge in Alternative Mengeneinheit umrechnen ---------------\**  
      CALL FUNCTION 'UNIT\_CONVERSION\_WITH\_FACTOR'  
           EXPORTING  
                ADD\_CONST   = ADDK  
                DENOMINATOR = UMREN  
                INPUT       = XMENGE  
                NUMERATOR   = UMREZ  
           IMPORTING  
                OUTPUT      = XMENGE  
           EXCEPTIONS  
                OVERFLOW    = 2.  
      CASE SY-SUBRC.  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW.  *"   Feldüberlauf*  
      ENDCASE.  
      O\_ERFMG = YMENGE.                *"Umgerechnete Menge*  
      O\_MENGE = XMENGE.                *"Eingegebene  Menge*  
      O\_ERFME = ERFME.  
      O\_MEINS = MEINS.  
*\*     note 1158817: avoid dump BCD\_FIELD\_OVERFLOW*  
*\*      O\_UMREZ = UMREZ.*  
*\*      O\_UMREN = UMREN.*  
      describe field o\_umrez type t.  
      IF T EQ 'P'.  
        L\_UMREZ\_F = UMREZ.  
        L\_UMREN\_F = UMREN.  
        CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
          EXPORTING  
              NOMIN               = L\_UMREZ\_F  
              DENOMIN             = L\_UMREN\_F  
          IMPORTING  
              NOMOUT              = \*MARM-UMREZ  
              DENOMOUT            = \*MARM-UMREN  
          EXCEPTIONS  
              CONVERSION\_OVERFLOW = 1.  
        IF SY-SUBRC = 0.  
          O\_UMREZ = \*MARM-UMREZ.  
          O\_UMREN = \*MARM-UMREN.  
        ELSE.  
          MESSAGE E037(BM) RAISING OVERFLOW.  
        ENDIF.  
      ELSE.  
        O\_UMREZ = UMREZ.  
        O\_UMREN = UMREN.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
END-ENHANCEMENT-SECTION.  
*\*$\*$-Start: MB\_UNIT\_CONVERSION\_03---------------------------------------------------------------$\*$\**  
ENHANCEMENT 3  /SAPMP/PIECEBATCH\_MB\_UNIT\_C000.    *"active version*  
  IF ( UMREN IS INITIAL OR UMREZ IS INITIAL ) OR  
*\* Mill 0024 Single Unit Batch SW*  
         LF\_FRACT5\_CONVERSION IS INITIAL OR  
         lf\_use\_batch\_factors = 'X'.  
    IF MATNR IS INITIAL.               *"Keine Materialnummer*  
      IF XKZMEINH IS INITIAL.          *"Umzurechnende Menge in LME*  
*\*- Umrechnungsfaktor(en) sind nicht gesetzt Materialnummer nicht vorh.-\**  
        CALL FUNCTION 'UNIT\_CONVERSION\_SIMPLE'  
             EXPORTING  
                  INPUT                = XMENGE  
                  NO\_TYPE\_CHECK        = SPACE  *"Typenprüfung erfolgt*  
                  UNIT\_IN              = MEINS  
                  UNIT\_OUT             = ERFME  
             IMPORTING  
                  ADD\_CONST            = ADDK  
                  DENOMINATOR          = UMREZ\_GES  
                  NUMERATOR            = UMREN\_GES  
                  OUTPUT               = XMENGE  
             EXCEPTIONS  
                  CONVERSION\_NOT\_FOUND = 1  
                  OVERFLOW             = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen unmögl.*  
          WHEN 2.  
            MESSAGE E106 RAISING OVERFLOW.   *"Feldüberlauf*  
        ENDCASE.  
  
    CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
         EXPORTING  
              NOMIN               =  UMREN\_GES  
              DENOMIN             =  UMREZ\_GES  
         IMPORTING  
              NOMOUT              =  UMREN\_DEC  
              DENOMOUT            =  UMREZ\_DEC  
         EXCEPTIONS  
              CONVERSION\_OVERFLOW = 1  
              OTHERS              = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E106 RAISING OVERFLOW.   *"Feldüberlauf*  
          WHEN 2.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen unmögl.*  
        ENDCASE.  
  
        O\_ERFMG = XMENGE.              *"623071*  
        O\_MENGE = YMENGE.              *"623071*  
        O\_ERFME = MEINS.  
        O\_UMREZ = UMREZ\_DEC.  
        O\_UMREN = UMREN\_DEC.  
      ELSE.                            *"Umzurechnende Menge in AME*  
        CALL FUNCTION 'UNIT\_CONVERSION\_SIMPLE'  
             EXPORTING  
                  INPUT                = XMENGE  
                  NO\_TYPE\_CHECK        = SPACE  *"Typenprüfung erfolgt*  
                  UNIT\_IN              = ERFME  
                  UNIT\_OUT             = MEINS  
             IMPORTING  
                  ADD\_CONST            = ADDK  
                  DENOMINATOR          = UMREZ\_GES  
                  NUMERATOR            = UMREN\_GES  
                  OUTPUT               = XMENGE  
             EXCEPTIONS  
                  CONVERSION\_NOT\_FOUND = 1  
                  OVERFLOW             = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrech. unmögl.*  
          WHEN 2.  
            MESSAGE E106 RAISING OVERFLOW.          *"Feldüberlauf*  
        ENDCASE.  
    CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
         EXPORTING  
              NOMIN               =  UMREN\_GES  
              DENOMIN             =  UMREZ\_GES  
         IMPORTING  
              NOMOUT              =  UMREN\_DEC  
              DENOMOUT            =  UMREZ\_DEC  
         EXCEPTIONS  
              CONVERSION\_OVERFLOW = 1  
              OTHERS              = 2.  
        CASE SY-SUBRC.  
          WHEN 1.  
            MESSAGE E106 RAISING OVERFLOW.   *"Feldüberlauf*  
          WHEN 2.  
            MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen unmögl.*  
        ENDCASE.  
        O\_ERFMG = YMENGE.              *"Umgerechnete Menge*  
        O\_MENGE = XMENGE.              *"Eingegebene  Menge*  
        O\_ERFME = MEINS.  
        O\_UMREZ = UMREZ\_DEC.  
        O\_UMREN = UMREN\_DEC.  
      ENDIF.  
    ELSE.  
*\*-- Materialnummer vorhanden Umrechnungsfaktor(en) sind nicht gesetzt--\**  
*\*-- KZMEINH Umrechnung in LME oder AME --------------------------------\**  
      CALL FUNCTION 'MATERIAL\_UNIT\_CONVERSION'  
           EXPORTING  
                INPUT                = XMENGE  
                KZMEINH              = XKZMEINH  
                MATNR                = MATNR  
                CHARGE               = CHARG  
                WERKS                = WERKS  
                MEINH                = ERFME  
                MEINS                = MEINS  
           IMPORTING  
                OUTPUT               = XMENGE  
           EXCEPTIONS  
                CONVERSION\_NOT\_FOUND = 1  *"Umrechnungsf. nicht bestimmt*  
                OVERFLOW             = 2  *"Feldüberlauf*  
                MATERIAL\_NOT\_FOUND   = 3  *"Material existiert nicht*  
                MEINS\_MISSING        = 4  *"Keine BasismE vorhaNDEN*  
                NO\_MEINH             = 5  *"Keine AlME angegeben*  
                OUTPUT\_INVALID       = 6  *"Menge ist kein Zahl*  
                MEINH\_NOT\_FOUND      = 7. *"ALME nicht zulässig*  
      CASE SY-SUBRC.  
        WHEN 1.  
         MESSAGE E101 RAISING CONVERSION\_NOT\_FOUND.*"Umrechnen nicht mgl.*  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW.  *"                 Feldüberlauf*  
        WHEN 3.  
         MESSAGE E103 RAISING  CONVERSION\_NOT\_FOUND.  
        WHEN 4.  
          MESSAGE E104 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 5.  
          MESSAGE E108 RAISING CONVERSION\_NOT\_FOUND.  
        WHEN 6.  
         MESSAGE E102 RAISING  CONVERSION\_NOT\_FOUND.  
        WHEN 7.  
          MESSAGE E107 RAISING CONVERSION\_NOT\_FOUND.  
      ENDCASE.  
*\*      DATA: ls\_lips  TYPE LIPS,                                "1611350 /2070407*  
*\*            lt\_lips  TYPE STANDARD TABLE OF lips.              "1611350*  
*\*      IMPORT p1 TO lt\_lips FROM MEMORY ID 'LIPS'.              "1611350*  
*\*      IF sy-subrc = 0.                                         "1611350*  
*\*        READ TABLE lt\_lips INTO ls\_lips                        "1611350*  
*\*              WITH KEY                                         "1611350*  
*\*                   MATNR = MATNR                               "1611350*  
*\*                   CHARG = CHARG.                              "1611350*  
*\*        IF  sy-subrc = 0.                                      "1611350*  
*\*          IF  XMENGE NE ls\_lips-lgmng.                         "1611350*  
*\*            XMENGE = ls\_lips-lgmng.                            "1611350*  
*\*          ENDIF.                                               "1611350*  
*\*        ENDIF.                                                 "1611350*  
*\*      ENDIF.                                                   "1611350*  
      IF XKZMEINH IS INITIAL.  
       O\_ERFMG = XMENGE.                *"Eingegebene  Menge*  
       O\_MENGE = YMENGE.                *"Umgerechnete Menge*  
      ELSE.  
       O\_ERFMG = YMENGE.                *"Umgerechnete Menge*  
       O\_MENGE = XMENGE.                *"Eingegebene  Menge*  
      ENDIF.  
  
       O\_ERFME = ERFME.  
       describe field o\_umrez type t.  
       IF T EQ 'P'.  
         L\_UMREZ\_F = UMREZ.  
         L\_UMREN\_F = UMREN.  
         CALL FUNCTION 'CONVERT\_TO\_FRACT5'  
           EXPORTING  
               NOMIN               = L\_UMREZ\_F  
               DENOMIN             = L\_UMREN\_F  
           IMPORTING  
               NOMOUT              = \*MARM-UMREZ  
               DENOMOUT            = \*MARM-UMREN  
           EXCEPTIONS  
               CONVERSION\_OVERFLOW = 1.  
         IF SY-SUBRC = 0.  
           O\_UMREZ = \*MARM-UMREZ.  
           O\_UMREN = \*MARM-UMREN.  
         ELSE.  
           MESSAGE E037(BM) RAISING OVERFLOW.  
         ENDIF.  
       ELSE.  
         O\_UMREZ = UMREZ.  
         O\_UMREN = UMREN.  
       ENDIF.  
  
      IF UMREN  = UMREZ.               *"Umrechnungsfaktoren gleich ?*  
        EXIT.                          *"Raus*  
      ENDIF.  
    ENDIF.  
  ELSE.  
*\*----- Umrechnungsfaktor(en) ist/sind gesetzt -------------------------\**  
    IF XKZMEINH IS INITIAL.            *"Umzurech. Menge in LME*  
*\*-------- Menge in Lagermengeneinheit umrechnen ----------------------\**  
      CALL FUNCTION 'UNIT\_CONVERSION\_WITH\_FACTOR'  
           EXPORTING  
                ADD\_CONST   = ADDK  
                DENOMINATOR = UMREZ  
                INPUT       = XMENGE  
                NUMERATOR   = UMREN  
           IMPORTING  
                OUTPUT      = XMENGE  
           EXCEPTIONS  
                OVERFLOW    = 2.  
      CASE SY-SUBRC.  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW. *"Feldüberlauf*  
      ENDCASE.  
      O\_ERFMG = XMENGE.                *"Umgerechnete Menge*  
      O\_MENGE = YMENGE.                *"Eingegebene  Menge*  
      O\_ERFME = ERFME.  
      O\_UMREZ = UMREZ.  
      O\_UMREN = UMREN.  
    ELSE.                              *"Umzurechnende Menge in AME*  
*\*-------- Menge in Alternative Mengeneinheit umrechnen ---------------\**  
      CALL FUNCTION 'UNIT\_CONVERSION\_WITH\_FACTOR'  
           EXPORTING  
                ADD\_CONST   = ADDK  
                DENOMINATOR = UMREN  
                INPUT       = XMENGE  
                NUMERATOR   = UMREZ  
           IMPORTING  
                OUTPUT      = XMENGE  
           EXCEPTIONS  
                OVERFLOW    = 2.  
      CASE SY-SUBRC.  
        WHEN 2.  
          MESSAGE E106 RAISING OVERFLOW.  *"   Feldüberlauf*  
      ENDCASE.  
      O\_ERFMG = YMENGE.                *"Umgerechnete Menge*  
      O\_MENGE = XMENGE.                *"Eingegebene  Menge*  
      O\_ERFME = ERFME.  
      O\_MEINS = MEINS.  
      O\_UMREZ = UMREZ.  
      O\_UMREN = UMREN.  
    ENDIF.  
  ENDIF.  
  
ENDENHANCEMENT.  
*\*$\*$-End:   MB\_UNIT\_CONVERSION\_03---------------------------------------------------------------$\*$\**  
  
ENDFUNCTION.